SAS® Software and the Single Research Scientist

Mic Lajiness

Introduction

In order to survive, the pharmaceutical industry must discover new chemical entities (NCE's), develop them and then bring them to market. The development and marketing process involves obtaining approval through the New Drug Application (NDA) procedure which has been well documented in previous SUGI proceedings (1,2,3). This paper intends to focus on the role of SAS® in the support of the drug discovery process at The Upjohn Company, a U.S.-based pharmaceutical company.

Since the early 1970's SAS® has been an important tool to workers in the pharmaceutical industry. The primary power of SAS® in the early years lay in its powerhouse of statistical procedures that quickly became a fixture in terms of NDA submissions. SAS® was also used to analyze so-called pre-clinical data but the data entry and management features of SAS® 72 limited its role. At that time SAS® was basically a statistical procedure library with some "other features". It was really not a major player in laboratory automation and research information-based projects. This would soon change.

It is now 1990 and SAS® has changed quite a bit. From a 255 page SAS® 72 manual we have "improved" SAS documentation to the tune of over 2,000 pages in the BASE and STATS manuals alone! This was not the only change. SAS® now offers a variety of software modules, an interactive matrix language (SAS/IML®), a family of full screen products (SAS/FSP®), sophisticated graphics software (SAS/GRAPH®), application development software (SAS/AF®) and many others. In particular it is felt that the combination of SAS/AF®, SAS/FSEDIT®, SAS/GRAPH® and of course base SAS® offers a unique combination of features that is indispensable to productivity enhancement in the pharmaceutical industry.

In what way is SAS® a major tool and how can it be used to maximum advantage? To answer this question it might be of value to examine a series of real applications currently used within the research division of The Upjohn Company. The research mainframe is an IBM 3090/ series 300 running VM/CMS. The remainder of this paper will focus on seven different applications that have involved the SAS® system. An overview and examination of the main features of each application will be presented. This view of a multitude of SAS® applications will provide an illustration of how SAS® can provide a variety of useful functions that meet the needs of scientists involved in the drug discovery process.

GENERIC

This system is one that meets a rather fundamental need within the research environment and was developed in the early 80's. That need is for a general way to quickly and easily get data into a central database system. The need at Upjohn was a method for scientists to enter summary data into the Cousin system with minimal effort while being relatively bullet-proof! Most research data are very similar -- scientists examine novel synthetic chemical compounds for a particular biological activity. Typical information may consist of 1) the registry number of the compound, 2) the date of testing, 3) notebook reference, 4) comments, and 5) biological activity value. Examples of biological values include ED50 values (an estimate of the dose required to give 50% of the desired biological effect) and percent inhibition of enzyme activity. Most of the data scientists wish to enter is generally the same, however, their experience level with computer equipment is usually minimal.

The approach used to answer this need was development of GENERIC, a REXX exec-based system that utilized general purpose SAS® routines. Use of REXX to control SAS® processing has been described earlier by Carpenter(4). In this application the SAS® and REXX routines provide the user a multitude of functions such as:

1) Ability to access Help information
2) Ability to access Entry/Editing screens
3) Ability to create entry systems for new assays
4) Ability to Update the COUSIN database.
5) Ability to send E-mail notes to the developer.
6) Ability to obtain SAS®-based reports

These functionalities are provided by creating modules of SAS® code (see figure 1) that are %INCLUDEd into the base SAS routines which then perform the desired functions (see figure 2). A flat file data storage method is used for two reasons: it allows users to examine the file from outside of SAS®, and it uses the file to update COUSIN directly.

The CAINFLUX System

CAINFLUX is a very simple system and seems a good place to start the examination of SAS® usage in preclinical research. The basic construction of the system is given in figure 1. A REXX exec is used to display a menu that allows the user to select one of six options. The options include 1) Getting help information, 2) Sending electronic mail to the developers, 3) Entry/Editing of the Master file, 4) Updating of the Cousin database, and 5) Terminating the exec. The REXX exec approach was chosen because one of the main functions of this system is to send a file to update a chemical information database called COUSIN (1). The program used to perform the updates will not run in CMS subset mode; it will not run from within a SAS® job.

Data is stored in permanent SAS® datasets locally and in an IBM SQL/DS table for use in the COUSIN system. The duplicate storage is intentional since the ability of SQL/DS to restore individual tables is very limited. In this particular system no statistical processing is performed at all -- summary data only is entered into the system via SAS/FSEDIT® software.

Another feature of this system is that the application itself resides on a CMS "nolog" account -- a CMS account that cannot be "logged on to" . READ/WRITE access is granted to the account on a need-to-update basis. This allows the developer to update the system without needing to "signon" the user's CMS account. While this approach would not work if simultaneous update functions are required, it does have some advantages over using SAS/SHARE® software. One advantage is that the user does not have to signoff one account and signon the other to access the application id. This application is actually used by researchers in Tsukuba, Japan. The storage of code and files on the nolog account is invaluable since it is very difficult to contact our Japanese researchers to fix problems and make changes.

In summary, the main features of this particular approach are that it is simple and fast to develop, and that it utilizes a "nolog" CMS id to balance security issues with the need for multiple user updates and ickiest to system files.

The HGASSAY System

The Hair Growth Assay system is a integrated data management, analysis and reporting system. The basic design of the system is illustrated in figure 3.

Periodic observations are recorded on the experimental animals over one to several weeks. The observations include hair growth in milligrams and body weight. Data is entered by a technician via a SAS/FSEDIT® screen into a raw data file. Throughout the study the scientist or assistant can request preliminary statistical analysis or examination of hair weight or body weight changes.

Prior to development of the present system a technician would build large flat files containing many hundreds of records in fixed fields. Subsequent examination of much of this data showed it was full of errors. The new system, which utilizes FSEDIT, allows for optimized data entry. It requires minimal entry for maximal benefit, with very few errors.

SAS/AF® produces the main menu from which all system options are accessed. Use of SAS/AF® in pre-clinical applications have been described in earlier proceedings (5,6). In the present system, there are several features of this system that warrant detailed examination. First is the generalized production of a histogram report as in figure 4. The report includes standard error bars along with N, Mean, and P-value from a comparison to control, and is self-adjusting to the number of treatment groups. Pseudo SAS® code illustrating the production of the histogram is given in figure 5. Another feature of this system is that essentially all statistical calculations are obtained indirectly from the SAS® statistical procedures. That is, the p-values and standard errors, etc are calculated directly from values obtained from a variety of sources (see figure 6). The reason for this is that the current SAS® statistical procedures do not allow full access to calculated values via OUTPUT datasets. One must be careful in routine application of statistical analyses. These are discussed in some detail by Altan, et al (7). PROC PRINTTO could be used to process the output (LISTING) file, but procedure output does change over time. The advantage of the indirect approach is that the input data relied upon for the calculations of p-values, for example, comes from procedures like PROC MEANS, and functions like PROBT. Estimates of pure error come from summing the squares of the residuals obtained from the analysis of variance.

This system also transfers information to the COUSIN database. In contrast to the previously described methods, this is accomplished in the HGASSAY System via an external routine that creates a flat file to update COUSIN.

NLIN2

NLIN2 is a customized nonlinear analysis routine that allows for input of raw data via machine or manual, simultaneous read/write access, fitting of a nonlinear model, prediction of EC50 value, and production of camera-ready graphics. The basic design of the system is given in figure 7. Data generally comes from a scintillation counter that creates a floppy disk which is then imported to the mainframe for processing, although manual entry is possible. Processing occurs when the NIN2 system is invoked and entry into the master files is controlled through a SAS/SHARE® connection which allows simultaneous read/write access (see figure 8) . The non linear modeling part is rather interesting because it provides the user with the automatic selection of the best of 3 models to obtain an IC50 estimate. Signal/noise ratio is used to determine which model "fits" best. Essentially the application performs expert system-style operations that minimize the amount of interaction with statisticians and produce the proper analyses. The production of high quality graphics (see figure 9) involves the generation of the estimated curve along with the overlay of the observed data points. It was found that simply running a spline or a SM-type smoothing routine through the predicted points was inadequate. This method frequently produced unrealistic fits due the relatively low number of predicted points. A better method involved using parameters obtained from the PROC NLIN routine and then generating a set number of points (e.g. 100) via the appropriate equation (see figure 10). Passing a simple spline through these points produces a perfect description of model behavior.

EASYNLIN

EASYNLIN is a system that provides for easy linear & non-linear modeling functions. The basic design of the system is given in figure 11. The main feature of the design is that users can choose from a 3x3 grid of graphical display of the various model options while simultaneously viewing the raw data in a central cell (see figure 12). Users can transform the original variables by a predefined set of choices and then iterate back through the model selection stage. Once the model is chosen, parameters for the model are derived from the data and the non-liner/linear analysis is performed. Reverse predictions, EC50 (point) estimations, or a variety of other functions can then be performed.

EASYNLIN utilizes SAS/AF®, SAS/GRAPH®(annotate), SAS/STAT® and SAS® Macros. An illustration of SAS pseudocode to produce the grid display is given in figures 13 and 14. The interesting feature in this method is that it uses a single macro routine to plot the form of any given model. Users provide only the cell location, model to be used, and min and max values. Pseudocode to perform the model comparison is given in figure 15.

SUMMARY

In summary, SAS®software has had a large impact on the productivity of scientists involved in pharmaceutical research and development. It should be clear from the above that one of the hallmarks of SAS® software is that it can be used to quickly develop reasonable solutions to computing needs. Further, SAS®software allows one the ability to rapidly implement the solution, and allow the developer to move on to other tasks. This particularly suits pharmaceutical research because it is not uncommon for screens or assays to be running one day and shut down the next. One cannot afford to devote a large amount of development time only to have the work discarded at the end.

ACKNOWLEDGEMENTS

The author would like to thank Nelson Pardee for his comments on the contents of this manuscript.

References

1. Rosenberg, M. Using the SAS®System to facilitate Clinical Trials Research and NDA Approval. 1988. SUGI13 Proceedings.

2. Rosenberg, M. An Integrated Approach to Computer Systems for NDA Preparation and Presentation. 1989. SUGI14 Proceedings.

3. Griffiths, G. Computer-aided NDA Review - a Microcomputer Approach. 1988. SUGI13 Proceedings.

4. Carpenter, A. Generic SAS® Programs: User Control Using REXX. 1989. SUGI14 Proceedings.

5. Bemis, K. Using SAS/AF* Software to Develop Preclinical Information Systems. 1988. SUGI13 Proceedings.

6. Bradshaw, M. The Selling of SAS® Software: Towards a Unified Analysis Environment in Basic Research. 1988.

SUGI13 Proceedings.

7. Altan, S., Devlin, T. PBS: An Intelligent Statistical System for Laboratory Scientists. 1988. SUGI13 Proceedings.

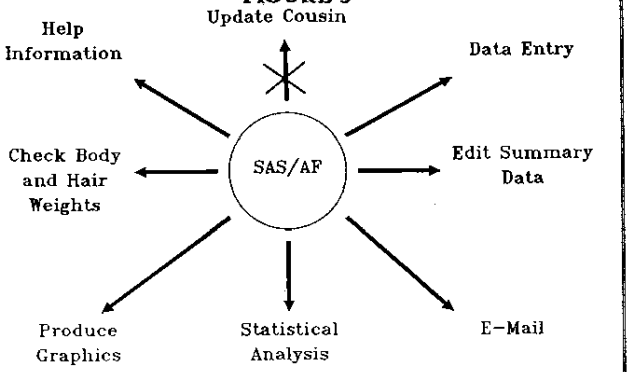
CONTACT

For reprints or more information please contact:

Mic Lajiness
Computational Chemistry
7247/267/1
The Upjohn Company
Kalamazoo, Michigan 49001
1-616-385-7494

Hair Growth Assay System

FIGURE 3

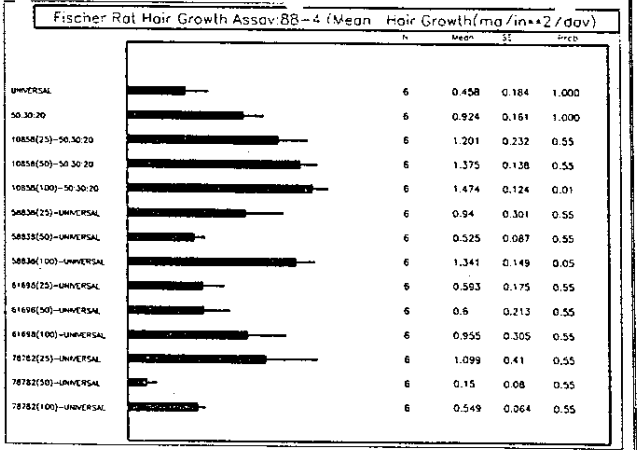


Data Definition Files

FIGURE 1

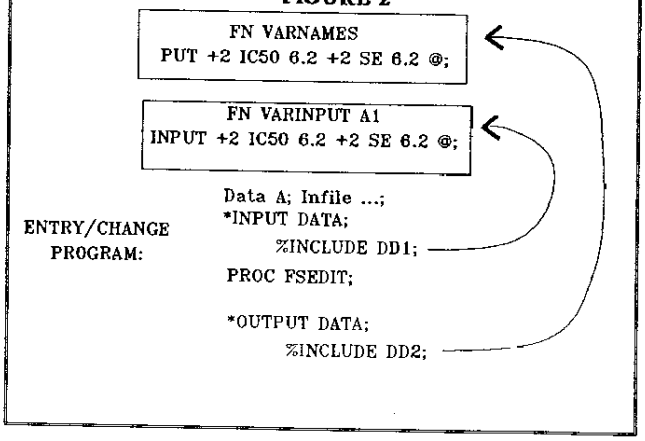
- VARINPUT – list of unique variable names defining input format
- VARNAMES – list of variables defining output format

FIGURE 4



Use of "% Includes"

FIGURE 2



Horizontal Histograms with SE bars

(Annotate macros)

```
DATA ANNO; IF _N_=1 THEN SET STATS; SET A;
IF _N_=1 THEN DO; %DCLANNO; %SYSTEM(5,5,5);..... END;
      COUNTER+1;
      Y1=(COUNTER*77/NTRT)+3; *----------Y CENTER OF RESP;
      X1=20;          X2=((MEAN/(MAX))*30)+X1;
      Y1=Y1+.8   ; Y2=Y1-1.6;
%BAR(X1,Y1,X2,Y2,RED,0,S);          * ---------- RESPONSE BAR;
      X2=(((MEAN+SE)/MAXX)*30)+X1; Y1=Y1-.8;
%LINE(X1,Y1,X2,Y1,RED,1,.5); * ----------SE OF RESPONSE;
      Y1=Y1+.6;
%LABEL(65,Y1,PUT(N,2.),WHITE,0,0,2.2,SIMPLEX,6);
%LABEL(73,Y1,PUT(MEAN,5.3),WHITE,0,0,2.2,SIMPLEX,6);
%LABEL(81,Y1,PUT(SE,5.3),WHITE,0,0,2.2,SIMPLEX,6);
%LABEL(89,Y1,PUT(PROB2,5.3),WHITE,0,0,2.2,SIMPLEX,6);
```

FIGURE 5

to Proc PRINTTO or not to Proc PRINTTO.....

FIGURE 6

```
        * GET SQUARED RESIDUALS
PROC GLM; BY TREAT; CLASSES TREAT; MODEL DAY=TREAT;
LSMEANS TREAT / STDERR PDIFF; OUTPUT OUT=OUT P=PRED R=RESID;
        * CALCULATE SSE AND N;
DATA OUT; SET OUT END=END;
TOTOBS+1; SSE+VAR*(RESID**2);
IF END THEN OUTPUT; RETURN;
        * CALCULATE THE FINAL STATS;
DATA A; SET OUT;
   DFERR=(TOTOBS-1) - (NUMGRPS-1);
   SE=SQRT(SSE/DFERR);
   DIFF=MEAN-CTRLMEAN;
   T=DIFF/SQRT(CTRLSE**2 + SE**2);
   PROB=(1-PROBT(ABS(T),DFERR))*2;
```

FIGURE 9

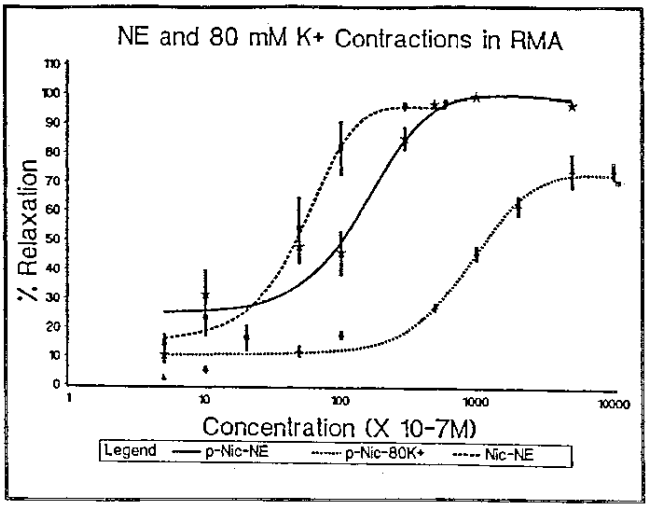


FIGURE 7

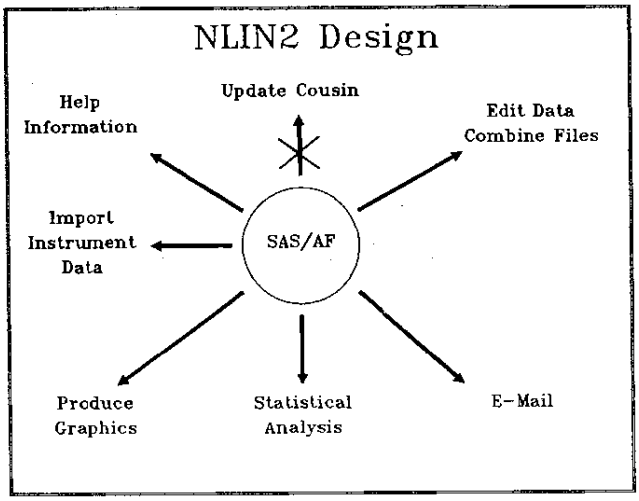


FIGURE 10

Graphing Data vs Model behavior Solution

- Capture model form:
  %LET MODEL=1/(B0+(B1)*EXP(-(B2)*X));
- Generate 100 points
  DO X=MIN to MAX by (MAX-MIN/100);
  PRED= &MODEL;
  OUTPUT;
  END;
- Spline

FIGURE 8

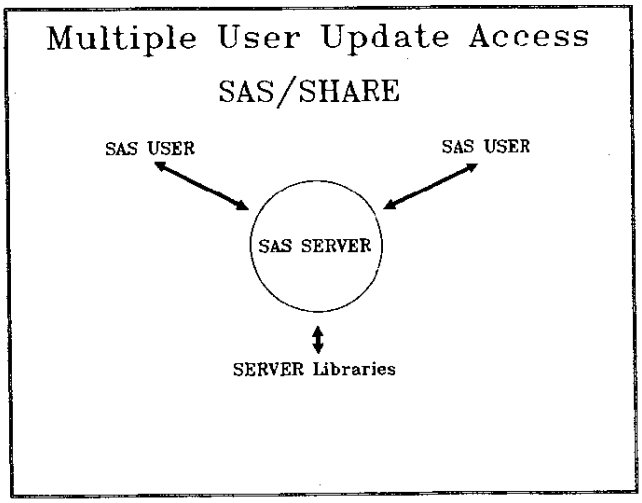


FIGURE 11

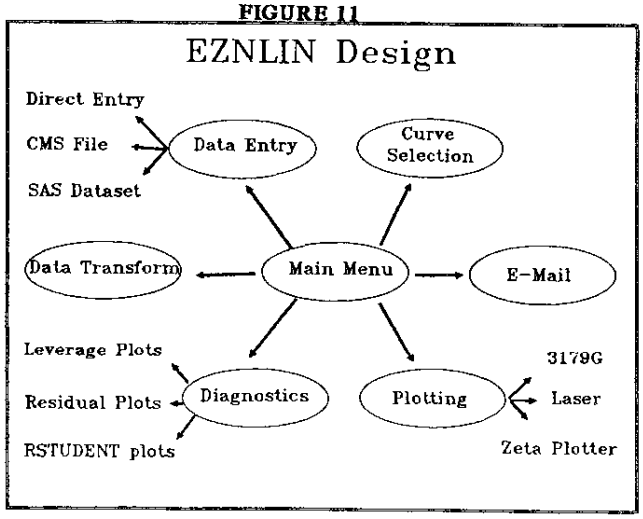


FIGURE 12

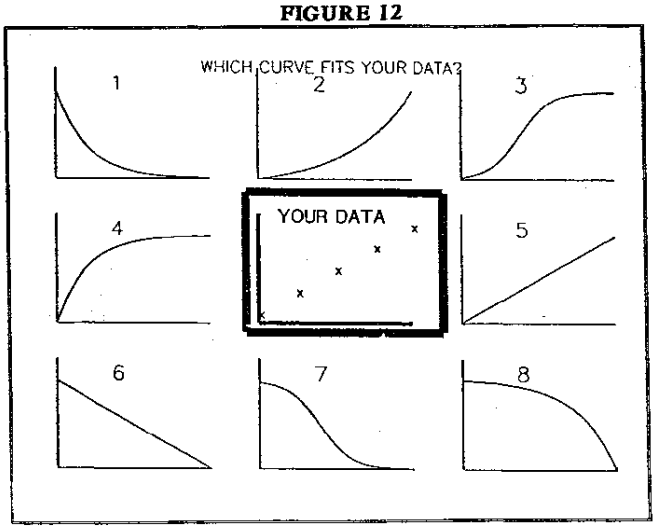


FIGURE 13

Model Display SAS® Code

```
%MACRO MODEL(FORM,XINIT,YINIT,XMIN,XMAX,YMIN,YMAX);
      %LET COUNT=%EVAL(&COUNT+1);          * SQUARE NUMBER;
      LX1=XINIT+10; LY1=YINIT+25;          * SQUARE LOCATION;
      %LABEL(LX1,LY1,"&COUNT",GREEN,0,0,5,SIMPLEX,E); *COUNT;
      %LINE(XINIT,YINIT,(XINIT+25),YINIT,WHITE,1,.2); *X-AX;
      %LINE(XINIT,YINIT,XINIT,(YINIT+25),WHITE,1,.2); *Y-AX;
      ITER=IMAX/60;                        * # OF ITERATIONS
      DO I=IMIN TO IMAX BY ITER;
         %STR(YVAL) = &FORM;               *FUNTION FORM;
         XVAL =25*((I-IMIN)/(IMAX-IMIN)) + XINIT;          *MAP X;
         YVAL =20*((YVAL-YMIN)/(YMAX-YMIN)) + YINIT; *MAP Y;
         IF I>IMIN THEN DO;
            %LINE(X1,Y1,XVAL,YVAL,RED,1,.2); END; *DRAW ;
         X1=XVAL;Y1=YVAL; END;* DEFINE PREVIOUS X & Y VALS;
%MEND;
```

FIGURE 14

Using the MODEL Macro

```
DATA ANNO;
%DCLANNO;
%SYSTEM(5,5,5);
   %MODEL(1+2*EXP(-.25*I),5,71,0,10,1,3);
   %MODEL(1/(2+10/I),5,38,0,10,.5,.34);
      .
      .
      .
   END;
PROC GANNO ANNO=ANNO;
```

FIGURE 15

Model Selection

```
%MACRO EZCOMP(FAMILY,MAX);
   %DO I = 1 %TO &MAX;
      DATA A&I; SET MODEL&FAMILY&I..DATA END=END;
      MODNUM="&FAMILY&I";
      IF RESID^=. THEN SUMRESID+RESID**2;
      IF RESID=. THEN SUMRESID=.;
      IF END THEN OUTPUT;
      %LET ALL = &ALL  A&I; * BUILD THE LIST OF DATASETS;
      %END;
   DATA A; SET &ALL END=END;
   PROC SORT; BY SUMRESID;
   DATA A; SET A; IF _N_=1;
   CALL SYMPUT('MODNUM',MODNUM);
%MEND;
```